

FIG. 1

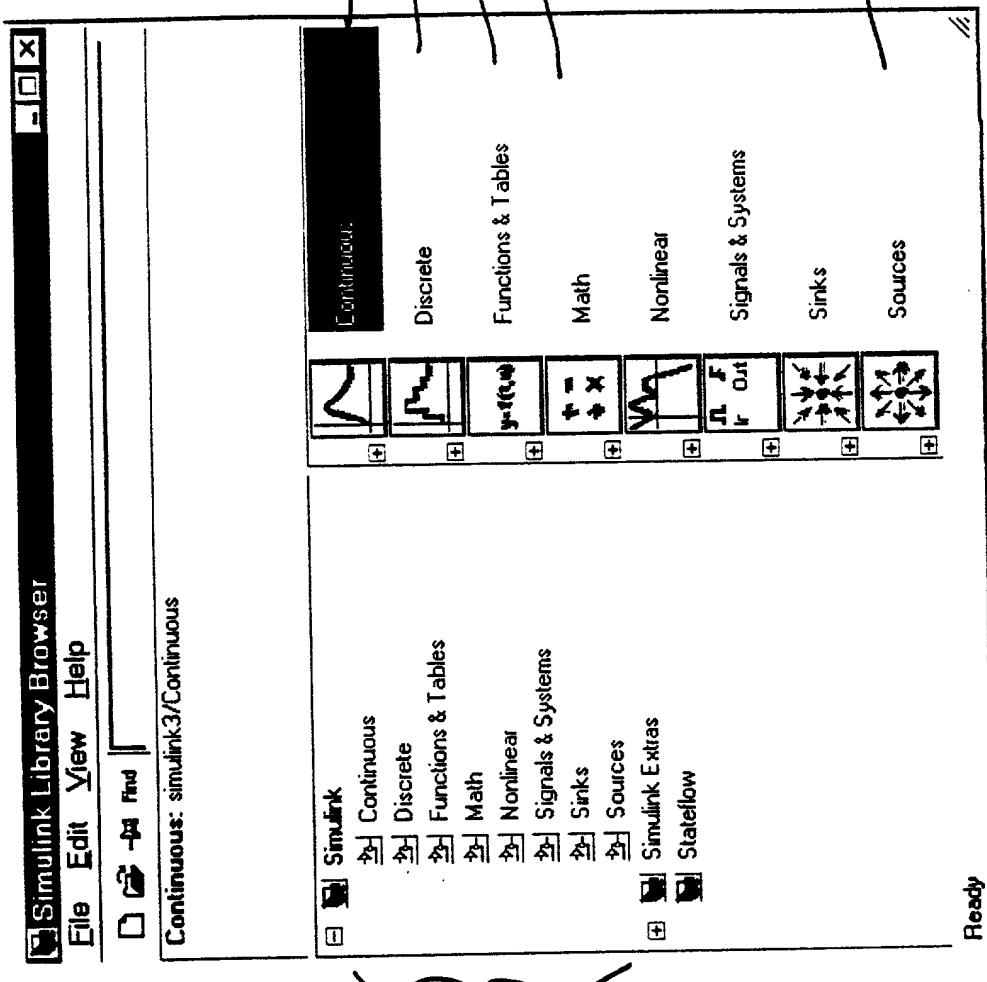


FIG. 2A

7.0 7.1 7.2 7.3 7.4 7.5 7.6 7.7 7.8 7.9

24

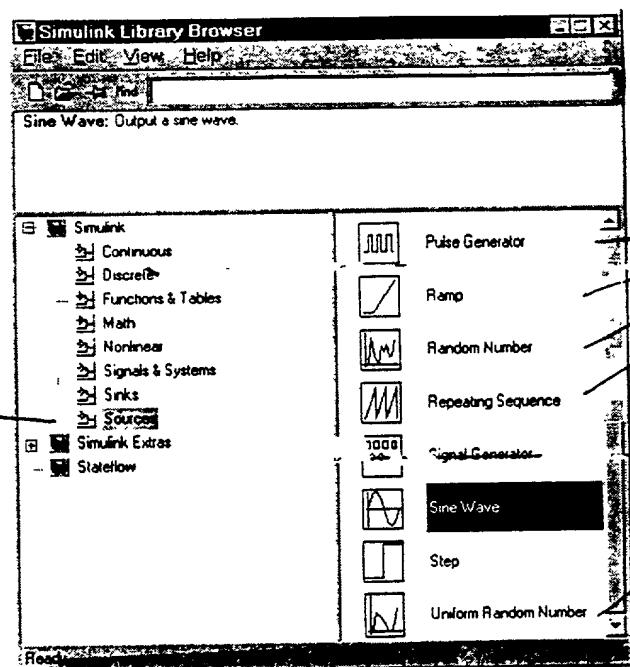


FIG. 2B

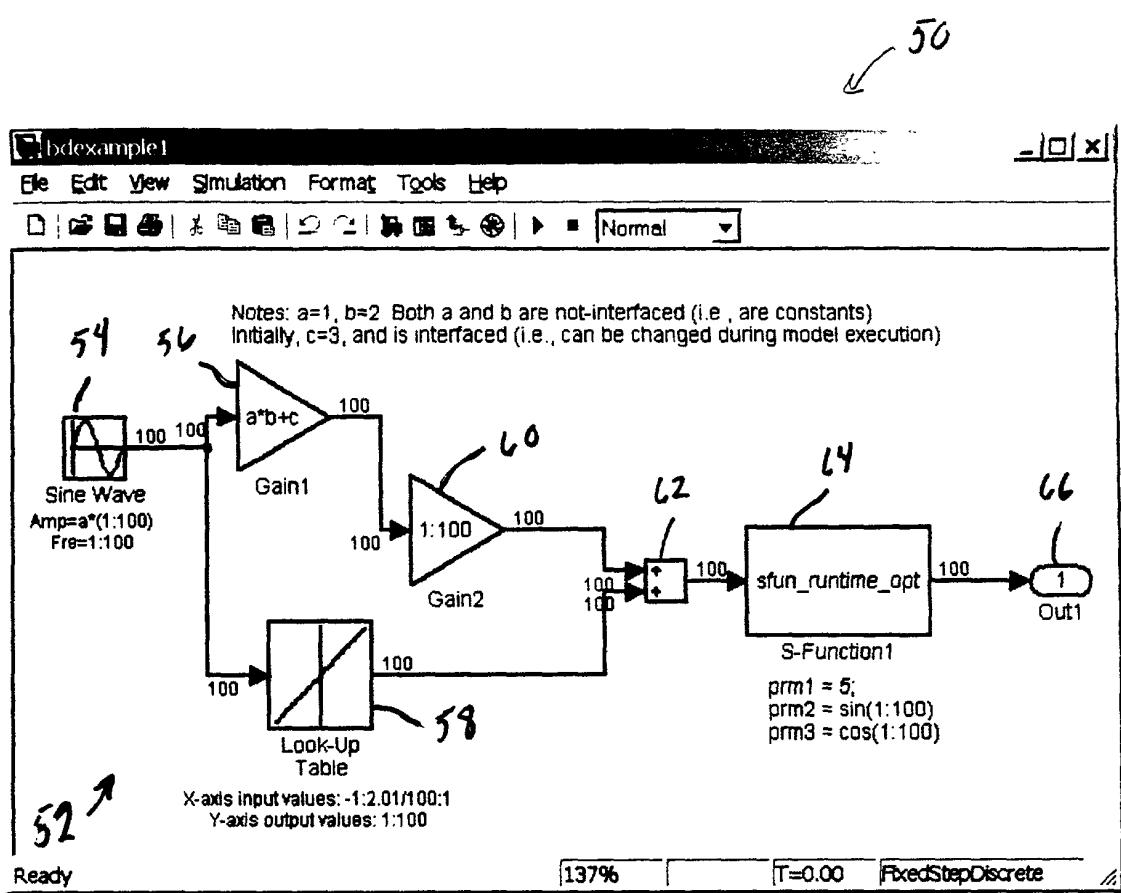


FIG. 3

70

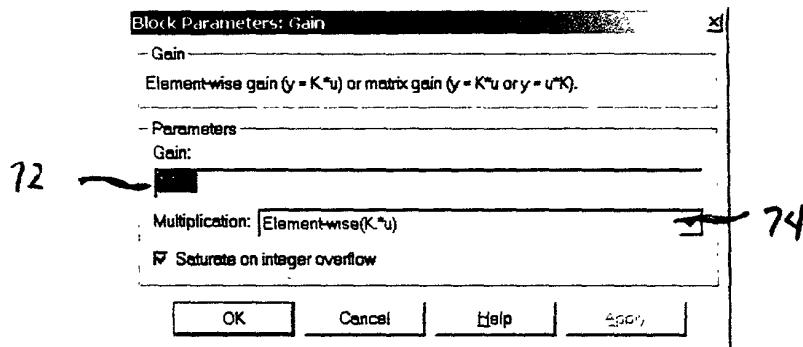


FIG. 4

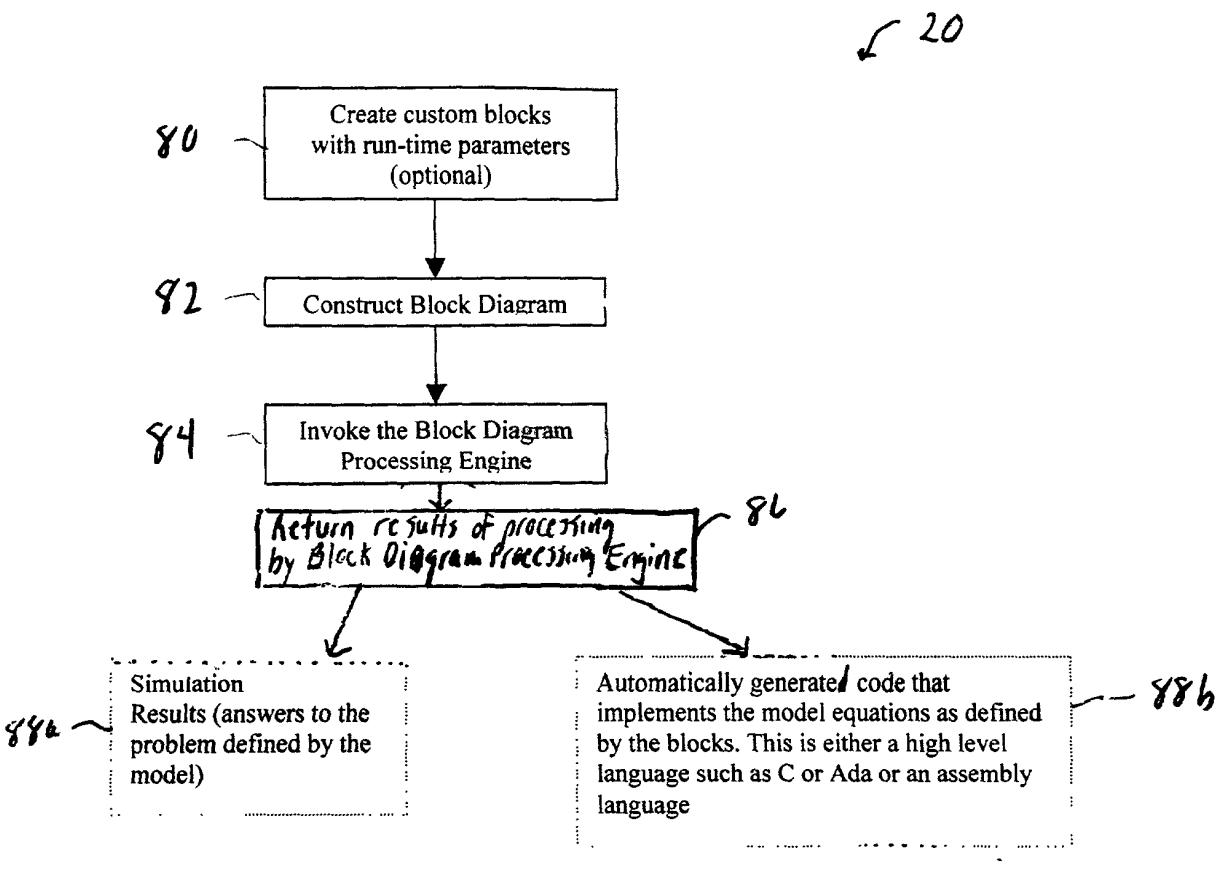


FIG. 5

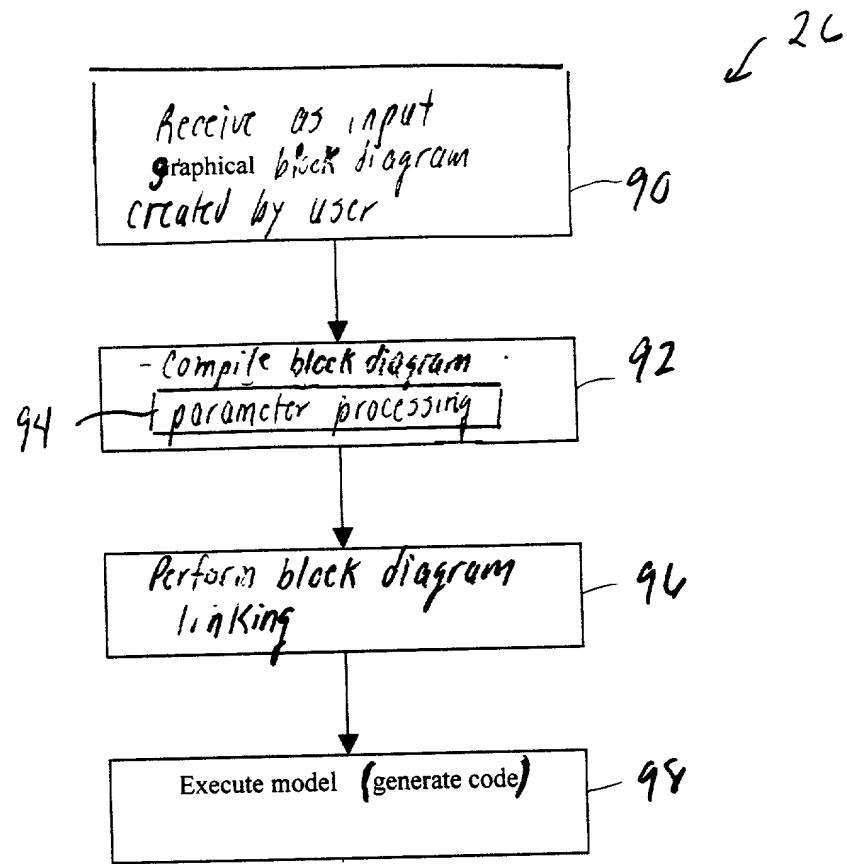


FIG. 6

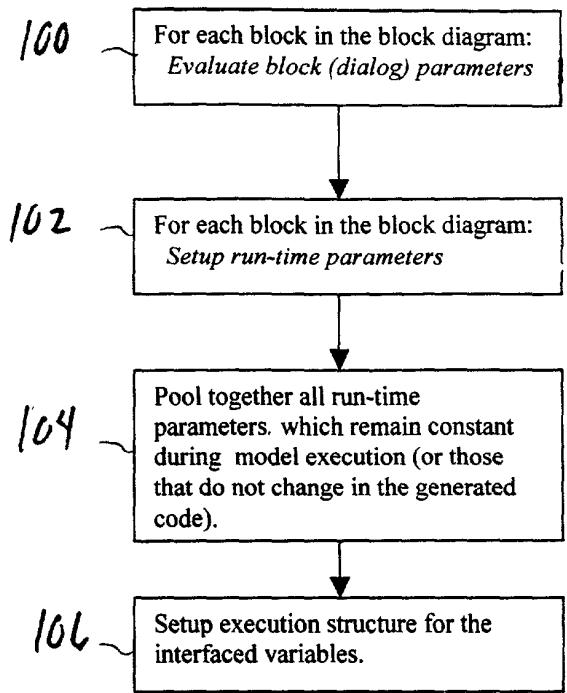


FIG. 7

110

```
Block {  
    EvalDialogParams(void) ← 112  
    SetupRuntimeParams(void) ← 114  
    InternalData: ← 116  
        AST[nParams]; ← 118  
        NumericDialogParamValues[nParams]; ← 120  
        RuntimeParams[nRunTimeParams]; ← 122  
    GetAstForDialogParam(int paramInt)  
    SetAstForDialogParam(int paramInt, ast)  
    GetNumericParamValues(int dialogParamIndex)  
    SetNumericParamValues(int dialogParamIndex, value)  
    GetNumRuntimeParams()  
    GetRuntimeParam(int runtimeParamIdx) ← 124  
    SetRuntimeParam(int runtimeParamIdx, value)  
}
```

FIG. 8

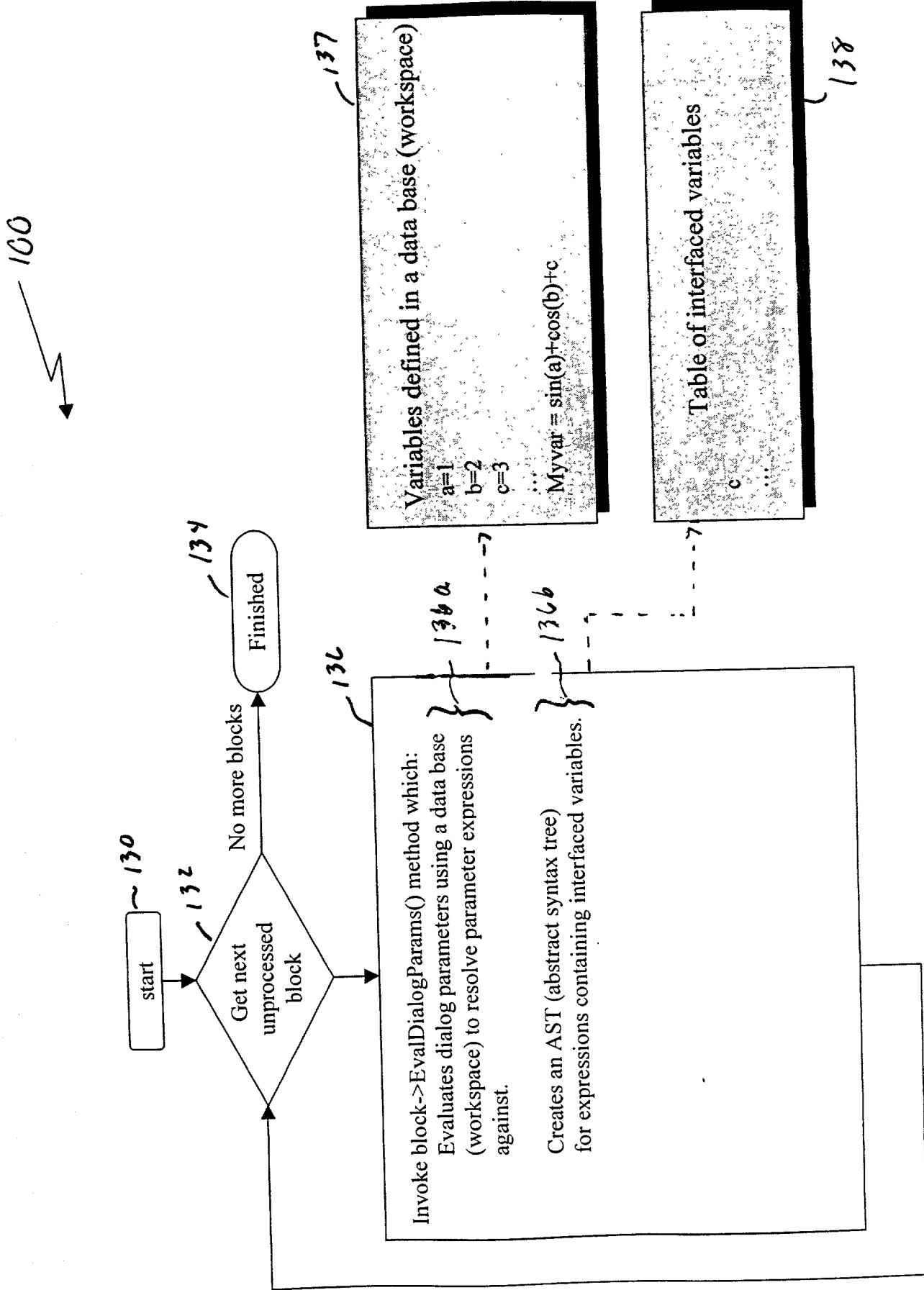


FIG. 9

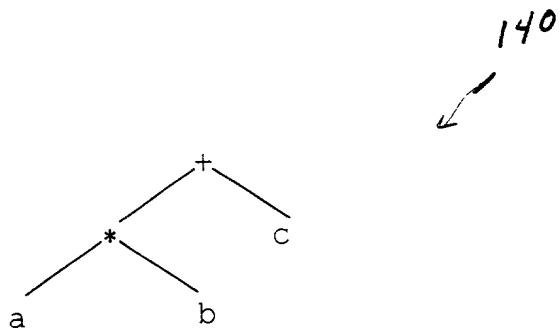


FIG. 10A

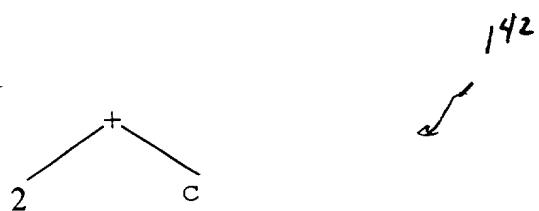
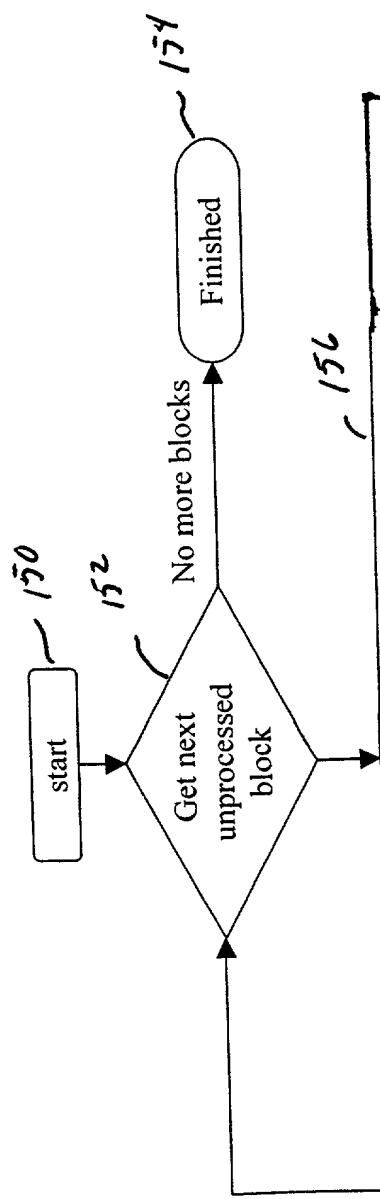


FIG. 10B



Invoke `block->SetupRuntimeParams()` method which :

Sets up the number of run-time parameters a block has. This is a function of the number of dialog parameters and the evaluated numeric values of the dialog parameters. This will appropriately size the `RuntimeParams[nRuntimeParams]` internal block array.

Create each run-time parameter. Each run-time parameter consists of its numerical value, a mapping back to the dialog parameters it was created from, and its characteristics, e.g., data type, numeric type (real vs. complex), whether or not it is transformed, and dimensions.

156a

156b

FIG. 11

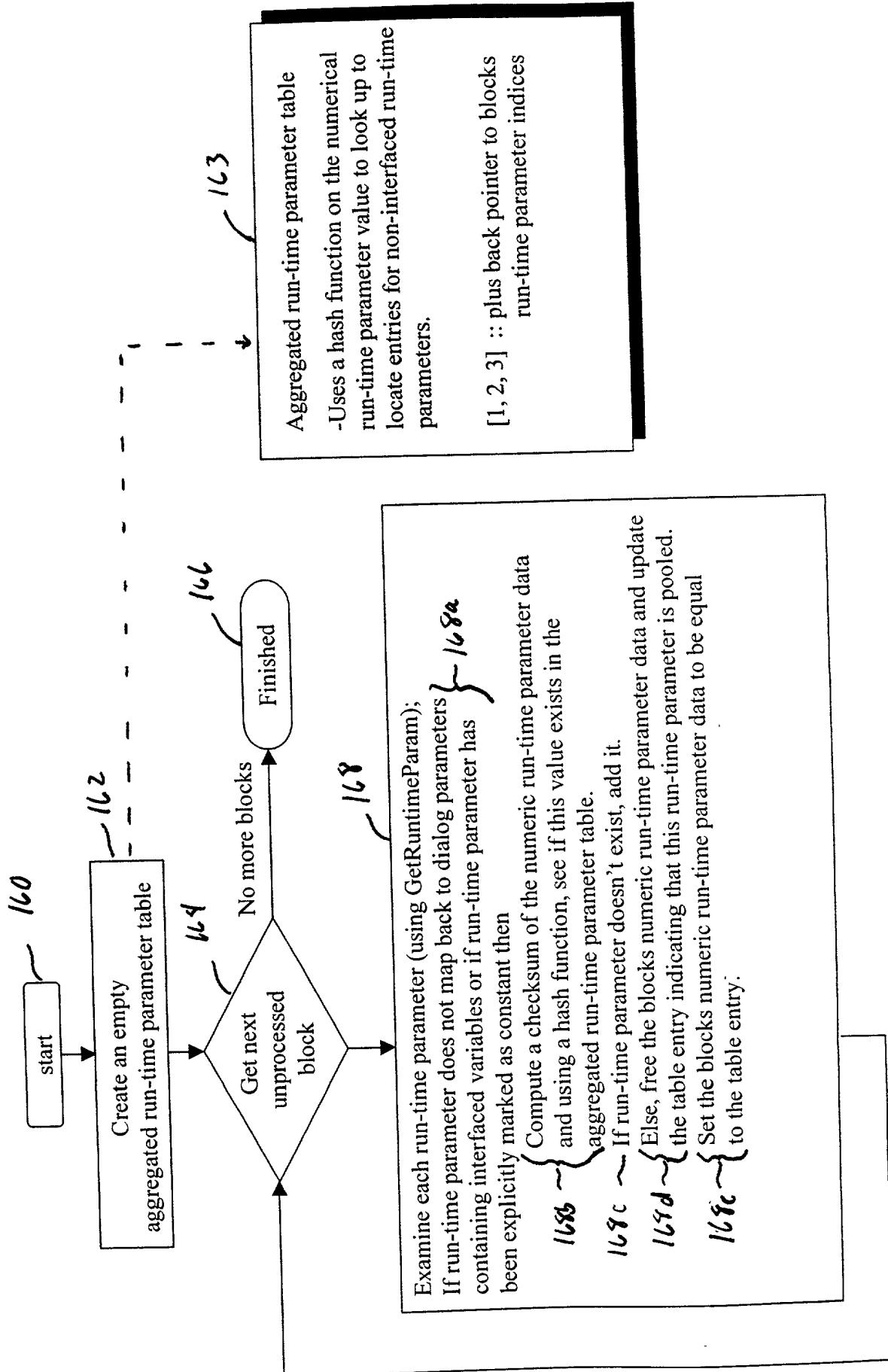


FIG. 12

FIG. 13

